## Énoncé

Vous devez développer une application pour un des sujets proposés suivants :

* **Bibliothèque** avec prêts de livres et location de films et/ou de jeux
* **Magasin** avec panier d'achats, facturation et gestion d'approvisionnement
* **Concessionnaire** automobile avec gestion d'inventaire
* **Entreprise** avec gestion de la paye des employés et contracteurs
* **Bancaire** avec gestion de comptes de chèque, d'épargne et de crédit
* **Clinique** avec gestion de rendez-vous
* **Zoo** avec gestion animaux et des enclos
* **Musée** avec gestion de vente de billets d'entrée

## À prendre en compte

L'ensemble du code de votre application devra couvrir la matière vue en classe tout au long de la session. L'idée du projet synthèse est de rassembler dans une même application tous les concepts de la programmation orientée objet qui ont étés vus dans le cours 420-2G2 Programmation orientée objet. Voici quelques éléments clés qui devront être présents dans votre application :

* De la modularisation (méthodes, avec ou sans paramètre) ;
* Des classes, des attributs et méthodes d’objet ;
* Des instanciations d’objets et des collections d’objets (listes d’objets) ;
* De l’héritage et des associations entre classes;
* De la sérialisation/désérialisation;
* Sans oublier les compétences de base en programmation (respect des normes de programmation, commentaires et documentation, gestion d'exception (validation d’entrée), etc.)

## Charge de travail

L’ampleur de ce travail n’est pas à prendre à la légère. Vous avez 3 semaines complètes pour développer l’ensemble du projet synthèse. Donc, un minimum de 33 heures (2 semaines x (7 heures en classe + 4 heures à la maison)) + 11 heures pendant la troisième semaine devraient être mis à l’élaboration du travail. Sachant que c’est tout de même une évaluation synthèse qui attestera ou non l’atteinte de la compétence, il se pourrait que vous ayez à mettre plus d’heures. Vous vous connaissez. Vous avez probablement déjà une idée si vous aurez besoin d’y mettre plus de temps. Le secret pour y arriver : une bonne organisation du travail.

## Auto-évaluation

À la fin de votre projet, vous devrez remplir un formulaire d’auto-évaluation selon les critères d’évaluation du projet.

## Démonstration/présentation

Vous devrez présenter votre projet synthèse durant la semaine d’examen. Cette présentation devra refléter votre auto-évaluation et couvrir les différents éléments qui vous seront mentionnés ultérieurement.

## Remises/livrables

Tout au long du développement de votre projet, des remises (des livrables) seront exigées et devront respecter un échéancier bien précis. Le tout devra être rassemblé pour la remise finale de votre projet. Toutes les remises se feront de façon électronique dans les Travaux sur LÉA.

|  |  |
| --- | --- |
| Livrables | Date de remise |
| Remise #0 – Estimation de la tâche | Vendredi 29 avril, minuit. |
| Remise #1 – Diagramme de classes UML complet et « presque » final | Mardi 3 mai, avant minuit |
| Remise #2 – Codage de l’UML (ensemble des classes) et leurs tests + Git | Vendredi 6 mai, minuit |
| Remise #3 – Interface graphique (maquette dont la mécanique est fonctionnelle) | Mardi 10 mai, avant minuit |
| Remise #4 – Remise complète de l’application (code complet) + Git | Jeudi 19 mai, minuit |
| Remise #5 – Auto-évaluation | Jeudi 19 mai, minuit (retards acceptés) |
| Présentation individuelle, dans la semaine d’examens (#6) | Pendant la semaine d’examen |

## Critères d’évaluation

|  |  |
| --- | --- |
| CRITÈRES | PONDÉRATION |
| Analyse adéquate du problème et  modélisation adéquate des classes (Remise #1) | 10% |
| Programmation adéquate des classes (Remise #2)  Programmation adéquate de l’ensemble de l’application  (Remise #4) | 75% |
| Génération d'interface graphique ergonomique et conviviale (Remise #3) | 5% |
| Documentation claire, juste et en conformité avec les normes de programmation du département  (Toutes les # Remises)  Utilisation appropriée du système de versionnage | 10% |